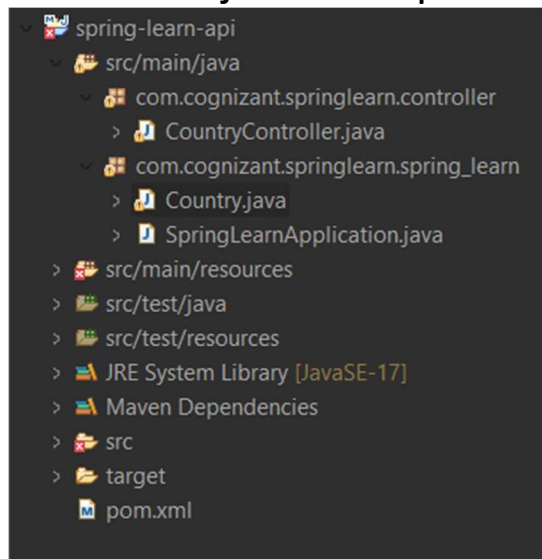# REST - Country Web Service

Maven Project in Eclipse

Updated pom.xml with dependencies

```xml
1  <project xmlns="http://maven.apache.org/POM/4.0.0"
2           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3           xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">
4      <modelVersion>4.0.0</modelVersion>
5      <groupId>com.cognizant.springlearn</groupId>
6      <artifactId>spring-learn-api</artifactId>
7      <version>0.0.1-SNAPSHOT</version>
8      <parent>
9          <groupId>org.springframework.boot</groupId>
10         <artifactId>spring-boot-starter-parent</artifactId>
11         <version>3.2.4</version>
12         <relativePath/>
13     </parent>
14     <properties>
15         <java.version>17</java.version>
16     </properties>
17     <dependencies>
18         <dependency>
19             <groupId>org.springframework.boot</groupId>
20             <artifactId>spring-boot-starter-web</artifactId>
21         </dependency>
22         <dependency>
23             <groupId>org.springframework.boot</groupId>
24             <artifactId>spring-boot-starter</artifactId>
25         </dependency>
26         <dependency>
27             <groupId>org.springframework.boot</groupId>
28             <artifactId>spring-boot-starter-logging</artifactId>
29         </dependency>
30     </dependencies>
31     <build>
32         <plugins>
33             <plugin>
34                 <groupId>org.springframework.boot</groupId>
35                 <artifactId>spring-boot-maven-plugin</artifactId>
36             </plugin>
37         </plugins>
38     </build>
39 </project>
```

Added country.xml

```xml
 1  <?xml version="1.0" encoding="UTF-8"?>
 2  <beans xmlns="http://www.springframework.org/schema/beans"
 3         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 4         xsi:schemaLocation="http://www.springframework.org/schema/beans
 5             http://www.springframework.org/schema/beans/spring-beans.xsd">
 6
 7      <bean id="country" class="com.cognizant.springlearn.spring_learn.Country">
 8          <property name="code" value="IN"/>
 9          <property name="name" value="India"/>
10      </bean>
11
12  </beans>
13  |
```

Added SpringLearnApplication.java

```java
 1  package com.cognizant.springlearn.spring_learn;
 2
 3  import org.springframework.boot.SpringApplication;
 4  import org.springframework.boot.autoconfigure.SpringBootApplication;
 5
 6  @SpringBootApplication(scanBasePackages = "com.cognizant.springlearn")
 7  public class SpringLearnApplication {
 8      public static void main(String[] args) {
 9          SpringApplication.run(SpringLearnApplication.class, args);
10      }
11  }
```

Added CountryController.java

```java
 1  package com.cognizant.springlearn.controller;
 2
 3  import com.cognizant.springlearn.spring_learn.Country;
 4  import org.springframework.context.ApplicationContext;
 5  import org.springframework.context.support.ClassPathXmlApplicationContext;
 6  import org.springframework.web.bind.annotation.RequestMapping;
 7  import org.springframework.web.bind.annotation.RestController;
 8
 9  @RestController
10  public class CountryController {
11
12      @RequestMapping("/country")
13      public Country getCountryIndia() {
14          ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");
15          return context.getBean("country", Country.class);
16      }
17  }
```

Added country.java

```java
1  package com.cognizant.springlearn.spring_learn;
2
3  import java.io.Serializable;
4
5  public class Country implements Serializable {
6      private String code;
7      private String name;
8
9      public Country() {}
10
11     public String getCode() {
12         return code;
13     }
14
15     public void setCode(String code) {
16         this.code = code;
17     }
18
19     public String getName() {
20         return name;
21     }
22
23     public void setName(String name) {
24         this.name = name;
25     }
26
27     @Override
28     public String toString() {
29         return "Country [code=" + code + ", name=" + name + "]";
30     }
31 }
```

Created application.properties and updated with the local port number as 8086 to run

```
1 server.port=8086
2
```

Runned the main() method and the required output has been received on 8086 port number the web page

localhost:8086/country

Pretty-print ☐

{"code":"IN","name":"India"}